GPS Logger 需求说明

1. 实践环节描述

1.1 实践设计目的

通过让同学使用UML、C++语言、C#语言进行程序的设计和编写，是同学们进一步熟悉已经学习到的知识，进一步理解嵌入式的编程方式以及编程技巧。

1.2 实践设计适合人群

本次实验的主要适用的人群是有一定编程基础的学生，掌握基础的编程语言、数据结构，熟悉UML语言，了解基本的算法，有过小程序的编程经验。

1.3 实践设计描述

本次实验使用C语言进行嵌入式芯片上的程序开发，使用C#语言进行PC端的Windows图形界面的开发，嵌入式芯片与PC的连接采用USB通信协议。嵌入式芯片负责简单处理并记录GPS接收器接收到的信息，PC端负责对嵌入式芯片记录下来的信息进行进一步的处理，能够使嵌入式芯片记录下来的信息绘制成地图上的轨迹。

嵌入式系统单片机部分的程序实现接收GPS信息功能，接收到GPS卫星发出的信息之后，将信息进行解析，按照指定的格式存储到文本文件中，存储至SD卡中。文件格式需要能够被Windows操作系统所识别。

Windows图形用户界面的应用程序主要是通过USB端口和ATmage 128单片机所存储文件的SD卡进行交互。通过读取GPS文本信息，能够将文本中记录的信息在电子地图上以路径轨迹的方式显示出来。Windows图形用户界面还可以直接通过RS232协议同GPS接收机进行通讯，直接接收和存储GPS信息，并对接收到的信息进行解析、处理，实时的将GPS使用者的位置和路径显示在电子地图上。

2. 实践环节要求

2.1 硬件要求

嵌入式系统芯片采用 ATmega-128 芯片，PC 开发环境采用 Visual Studio 2008 或者 Visual Studio 2010，运行环境为 Intel-ATOM 嵌入式系统。GPS 接收机采用模拟的方式，信号由 GPS GATE 产生。

2.2 软件要求

嵌入式端程序要求采用 C 语言进行编写，PC 端程序采用 C 语言或者 C#语言进行编写。采用 UML 2.0 进行程序设计。

2.3 实践要求

功能要求：能够完整的完成需求说明书中的功能，并且经过基本的测试，做到无BUG发布。

文档要求：项目计划及实施文档、软件需求及设计文档、软件测试计划及测试用例文档、软件使用说明书。

测试要求：能够对编写的程序进行尽可能多的测试，包括：黑盒测试、白盒测试、灰盒测试等，对测试出现的BUG进行记录并予以修复。

2.4 其他要求

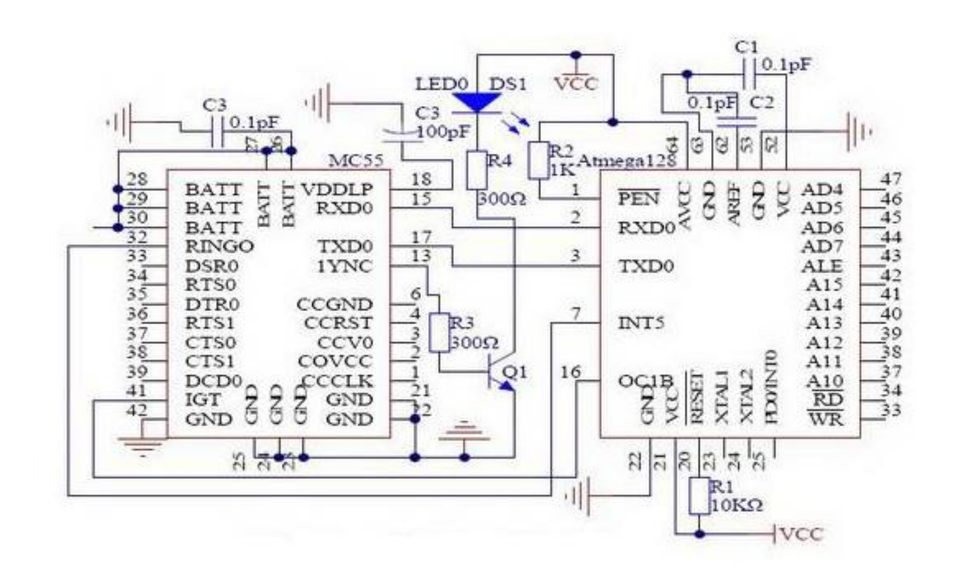
对于实践中不了解、不熟悉的问题要积极及时的同辅导老师、助教进行沟通。

3.开发环境简介

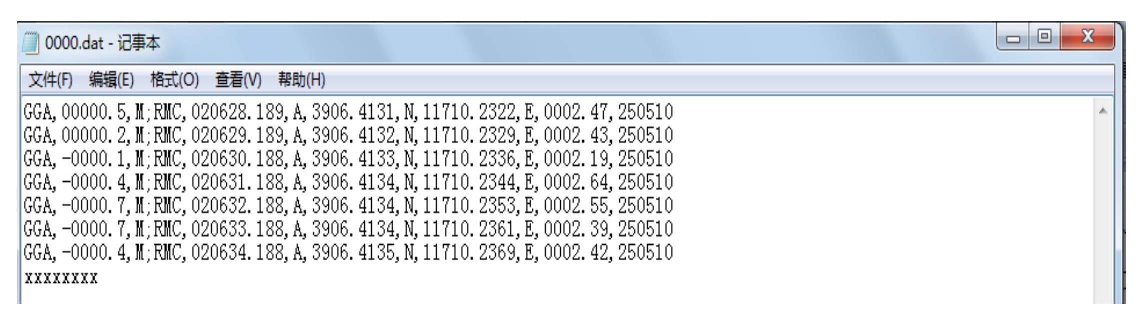
3.1 嵌入式部分

本次实践考虑到体积小、成本低的单片机比较容易控制，并且能够学习较多的嵌入式系统功能，因而采用了ATmega-128单片机。同时由于它在市场上的价格适中，并且拥有免费的开发工具以及海量的免费资源，利于同学们在平时、课外继续深入的研究嵌入式系统编程。本次编程采用的是AVR Studio 4集成环境和WinAVR C编译器进行编程。

实践的主要目的是掌握单片机的I/O端口，定时器/计数器，UART（通用异步串行接收发送器），SPI（同步串行接口），SD卡技术，FAT文件系统。通过与GPS接收机的交互，将接收到的GPS信息进行分析解析，通过UART传输到SD卡，以文本文件的形式存储起来。而记录下来的文件可以在PC的用户界面上进行还原，从而可以分析当时的状态和情况。



ATmega-128嵌入式芯片电路图


0000.dat - 记事本

文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

GGA,00000.5,M;RMC,020628.189,A,3906.4131,N,11710.2322,E,0002.47,250510
GGA,00000.2,M;RMC,020629.189,A,3906.4132,N,11710.2329,E,0002.43,250510
GGA,-0000.1,M;RMC,020630.188,A,3906.4133,N,11710.2336,E,0002.19,250510
GGA,-0000.4,M;RMC,020631.188,A,3906.4134,N,11710.2344,E,0002.64,250510
GGA,-0000.7,M;RMC,020632.188,A,3906.4134,N,11710.2353,E,0002.55,250510
GGA,-0000.7,M;RMC,020633.188,A,3906.4134,N,11710.2361,E,0002.39,250510
GGA,-0000.4,M;RMC,020634.188,A,3906.4135,N,11710.2369,E,0002.42,250510
xxxxxxxx

GPS文件记录结果截图

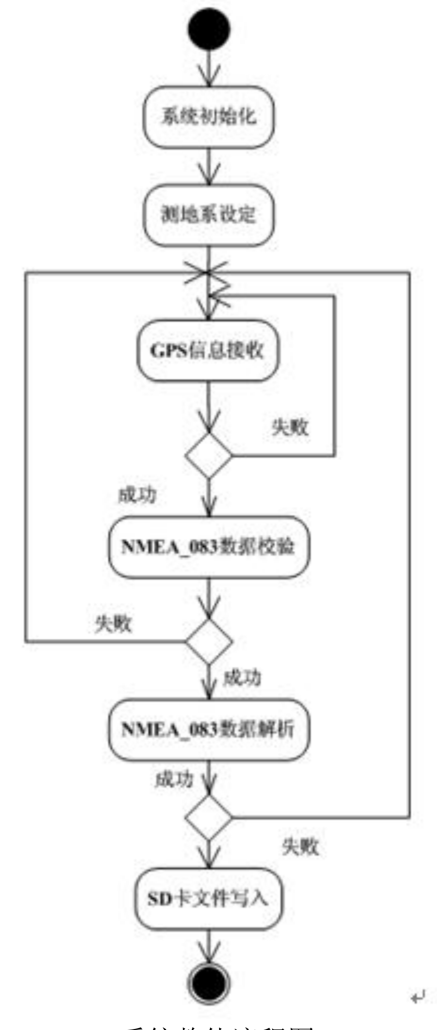
3.2 PC部分

本次实践的PC端的开发主要是巩固已经学到的编程语言或者试探性运用未学过的语言，因此开发环境是Windows操作系统，运用Visual Studio 2008或Visual Studio 2010进行用户界面的设计。经过测试的程序最终运行在Intel-ATOM嵌入式系统上。

4. 系统设计

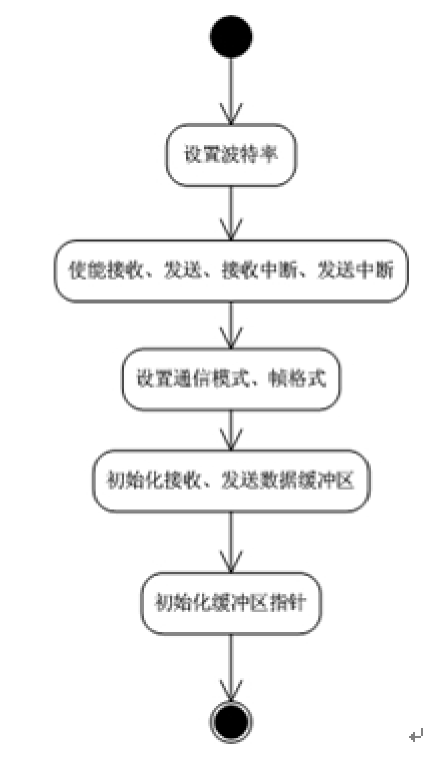
4.1 嵌入式部分程序设计

嵌入式部分程序主要分为三个部分，第一部分是嵌入式芯片接收GPS接收机发送的信息，并同时对这些信息进行分析、解析。第二部分是UART的设计，通过UART将信息传输到SD卡进行存储。第三部分是SD的信息存储部分，将信息存储成为Windows系统可以读取的文件格式。

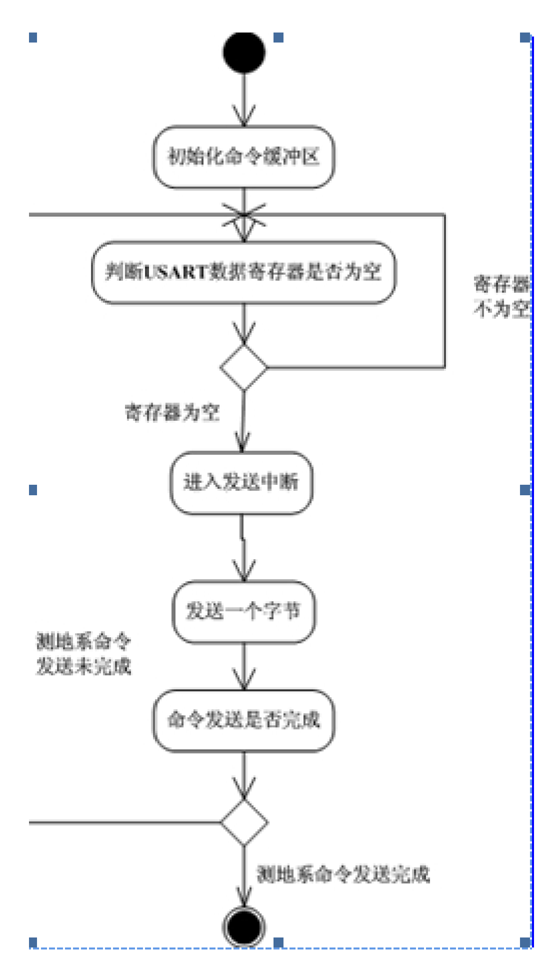


系统整体流程图

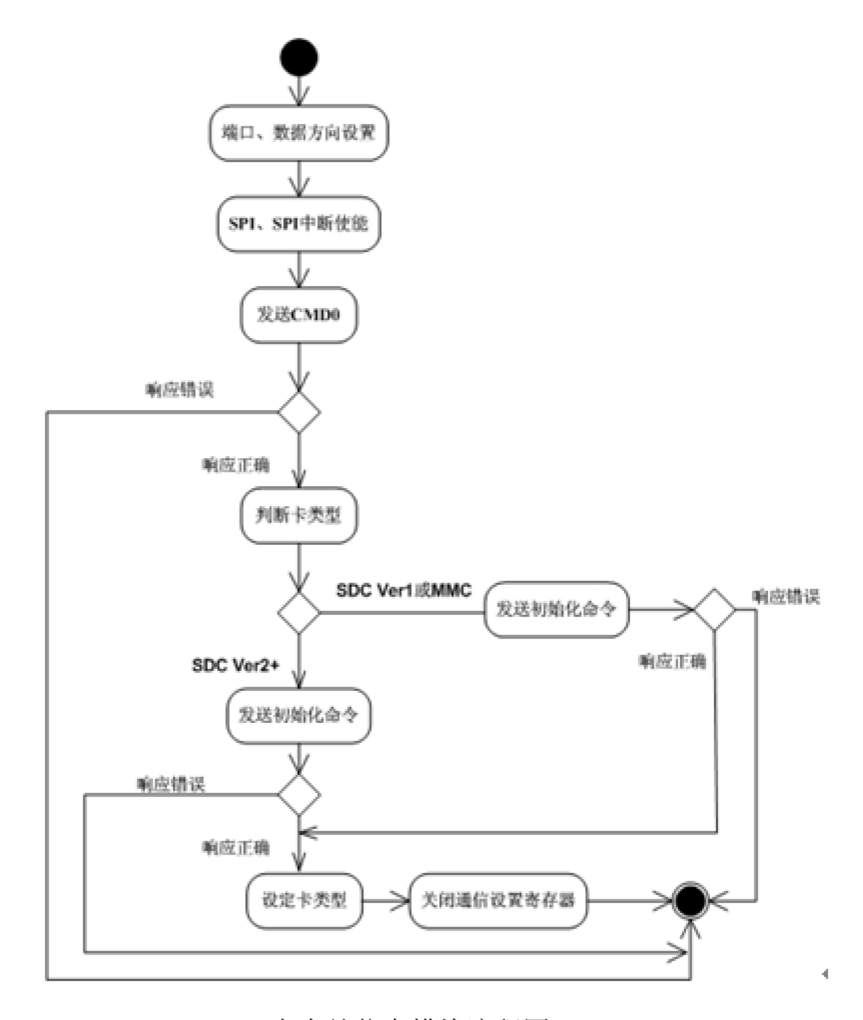
首先进行系统的初始化， 接着根据使用者的不同设定地系。当系统的初始化过程完成之后，开始进行GPS信息的接收，对接收到的信息则要进行数据校验，谨防出现坏数据，之后将受到的信息进行分析、解析，按照指定的格式将信息存储进入SD卡。下面有每个主要功能模块的流程图。



UART设计流程图



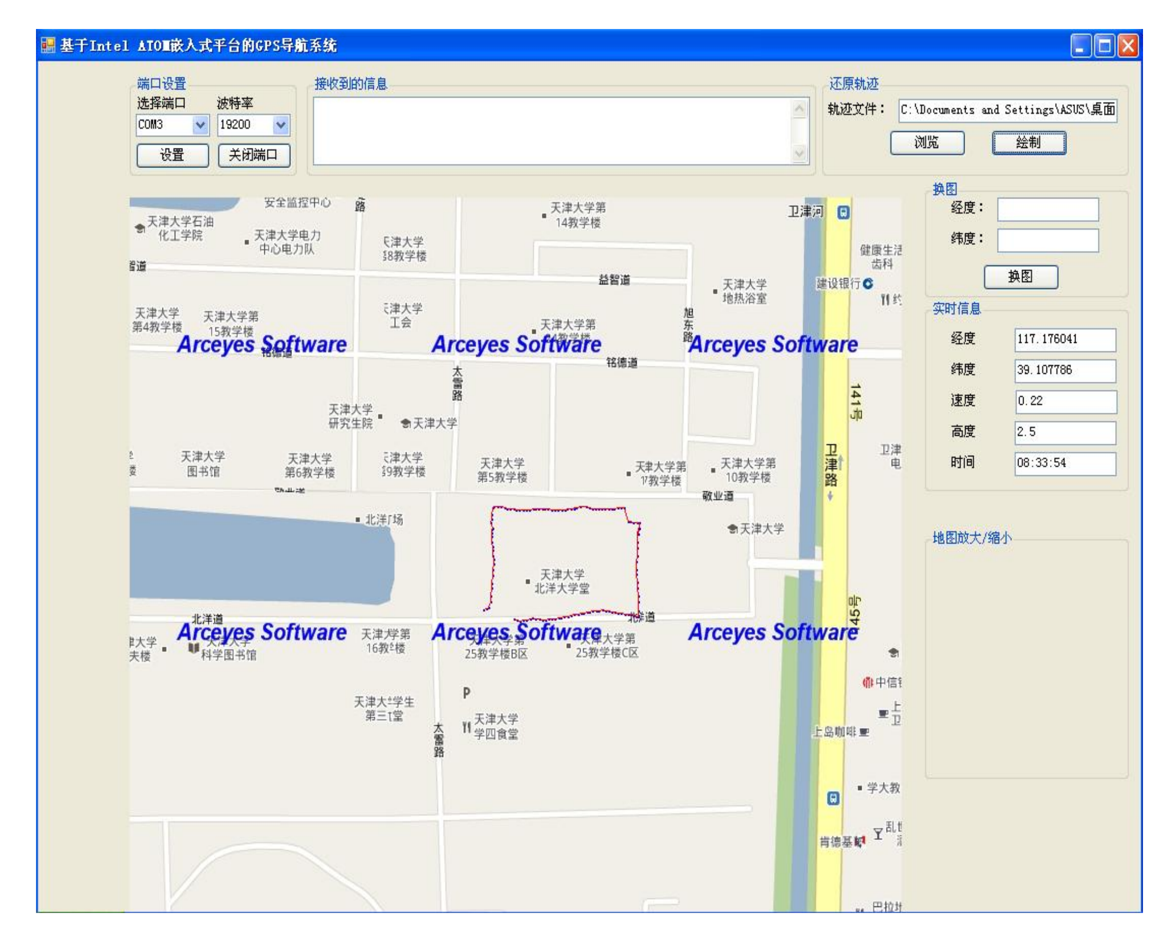
GPS信息接收模块流程图



SD卡存储信息模块流程图

4.2 PC部分程序设计

在PC部分的设计中，通过一个USB来读取存储在SD中的数据，系统负责将读取到的文件信息进行还原。通过加载第三方的免费电子地图，将记录的轨迹信息绘制在第三方地图上。



用户界面截图

5. 系统其它扩展

在嵌入式部分，虽然资源有限，但是在进行系统开发的时候，应该为以后功能的扩展预留接口，例如：A/D转换，PWM，实时操作系统等等。

在PC部分，学生可以添加地图放大缩小功能，提供最佳路径功能，提供旅游景点的预览图片功能等，作为学生的挑战项目。

学生也可以提出其它整体上的各种创新性的内容，来丰富这个实践。

6. 附录A

6.1 NMEA-0183格式的GPS信息

序　号	命　令	说　明	最大帧长
1	$GPGGA	全球定位数据	72
2	$GPGSA	卫星 PRN 数据	65
3	$GPGSV	卫星状态信息	210
4	$GPRMC	运输定位数据	70
5	$GPVTG	地面速度信息	34
6	$GPGLL	大地坐标信息	
7	$GPZDA	UTC 时间和日期	

GPS信息帧

下面介绍主要用到GPS信息：文中<CR><LF>代表十六进制的0x0d回车，0x0a换行。

1、 GPS DOP and Active Satellites（GSA）当前卫星信息

$GPGSA,<1>,<2>,<3>,<3>,,,,,,,,<3>,<3>,<3>,<4>,<5>,<6>,*<7><CR><LF>

<1>模式 ：M = 手动， A = 自动。

<2>定位型式 1 = 未定位， 2 = 二维定位， 3 = 三维定位。

<3>PRN 数字：01 至 32 表天空使用中的卫星编号，最多可接收 12 颗卫星信息。没有卫星信息的只保留逗号“，”。

<4> PDOP 位置精度因子（0.5~99.9）

<5> HDOP 水平精度因子（0.5~99.9）

<6> VDOP 垂直精度因子（0.5~99.9）

<7> *号后为 Checksum.(检查和). 字符’$’是 GPS 语句的起始标志，字符’*’是校验和的起始标志，字符’*’之后是一个两位的以 ASCII 格式显示的十六进制数字（即处于 00 到 FF 之间）。这两个字符表示成的十六进制数值就是该语句的校验和。语句校验和的计算方法为：将字符’$’与字符’*’之间的所有字符(不包括’$’和’*’)进行异或之后所得的结果。其他 GPS 信息的检查和计算方法相同。

GSA 信息例：

$GPGSA,A,3,05,24,17,30,02,,,,,,,,5.6,3.3,4.5*34<CR><LF>

2、 GPS Satellites in View（GSV）可见卫星信息

$GPGSV, <1>,<2>,<3>,<4>,<5>,<6>,<7>,?<4>,<5>,<6>,<7>,*<8><CR><LF>

<1> GSV 语句的总数

<2> 本句 GSV 的编号

<3> 可见卫星的总数，00 至 12。

<4> 卫星编号， 01 至 32。

<5>卫星仰角， 00 至 90 度。

<6>卫星方位角， 000 至 359 度。实际值。

<7>讯号噪声比（C/No）， 00 至 99 dB；无表未接收到讯号。

<8>Checksum.(检查位).

第<4>,<5>,<6>,<7>项个别卫星会重复出现,每行最多有四颗卫星。其余卫星信息会于次一行出现,若未使用，这些字段会空白。

GSV 信息例：

$GPGSV,3,1,12,30,72,254,30,05,70,125,39,24,37,083,43,02,36,113,45*7B<CR><LF>

3、Global Positioning System Fix Data（GGA）GPS 定位信息

$GPGGA,<1>,<2>,<3>,<4>,<5>,<6>,<7>,<8>,<9>,M,<10>,M,<11>,<12>*<13><CR><LF>

<1> UTC 时间，hhmmss（时分秒）格式

<2> 纬度 ddmm.mmmm（度分）格式（前面的 0 也将被传输）

<3> 纬度半球 N（北半球）或 S（南半球）

<4> 经度 dddmm.mmmm（度分）格式（前面的 0 也将被传输）

<5> 经度半球 E（东经）或 W（西经）

<6> GPS 状态：0=未定位，1=非差分定位，2=差分定位，6=正在估算

<7> 正在使用解算位置的卫星数量（00~12）（前面的 0 也将被传输）

<8> HDOP 水平精度因子（0.5~99.9）

<9> 海拔高度（-9999.9~99999.9）

<10> 地球椭球面相对大地水准面的高度

<11> 差分时间（从最近一次接收到差分信号开始的秒数，如果不是差分定位将为空）

<12> 差分站 ID 号 0000~1023（前面的 0 也将被传输，如果不是差分定位将为空）

<13> 检查和。

GGA 信息例：

$GPGGA,084238.528,5000.31671,N,04500.00004,E,1,5,0.0,0.0,M,0.0,M,,*51<CR><LF>

4、Recommended Minimum Specific GPS/TRANSIT Data（RMC）推荐定位信息

$GPRMC,<1>,<2>,<3>,<4>,<5>,<6>,<7>,<8>,<9>,<10>,<11>,<12>*<13><CR><LF>

<1> UTC 时间，hhmmss（时分秒）格式

<2> 定位状态，A=有效定位，V=无效定位

<3> 纬度 ddmm.mmmm（度分）格式（前面的 0 也将被传输）

<4> 纬度半球 N（北半球）或 S（南半球）

<5> 经度 dddmm.mmmm（度分）格式（前面的 0 也将被传输）

<6> 经度半球 E（东经）或 W（西经）

<7> 地面速率（000.0~999.9 节，前面的 0 也将被传输）

<8> 地面航向（000.0~359.9 度，以真北为参考基准，前面的 0 也将被传输）

<9> UTC 日期，ddmmyy（日月年）格式

<10> 磁偏角（000.0~180.0 度，前面的 0 也将被传输）

<11> 磁偏角方向，E（东）或 W（西）

<12> 模式指示（仅 NMEA0183 3.00 版本输出，A=自主定位，D=差分，E=估算，N=数据无效）

<13> 检查和。

RMC 信息例：

$GPRMC,084238.528,A,5000.31671,N,04500.00004,E,10.00,0.0,240511,0.0,E*62<CR><LF>

5、 Track Made Good and Ground Speed（VTG）地面速度信息

$GPVTG,<1>,T,<2>,M,<3>,N,<4>,K*<5><CR><LF>

<1> 以真北为参考基准的地面航向（000~359 度，前面的 0 也将被传输）

<2> 以磁北为参考基准的地面航向（000~359 度，前面的 0 也将被传输）,M 代表磁北方位

<3> 地面速度（000.0~999.9 节，前面的 0 也将被传输），N 代表节

<4> 地面速度（0000.0~1851.8 公里/小时，前面的 0 也将被传输），K 代表 km/h

<5> 检查和。

VTG 信息例：

$GPVTG,0.0,T,0.0,M,10.00,N,18.52,K*41<CR><LF>

7. 附录B

下面介绍 GPS GATE 的使用方法：

1. 首先安装 GPS GATE v.2.6.0.340，安装之后点击 Advanced Setup，进入图 1 界面。

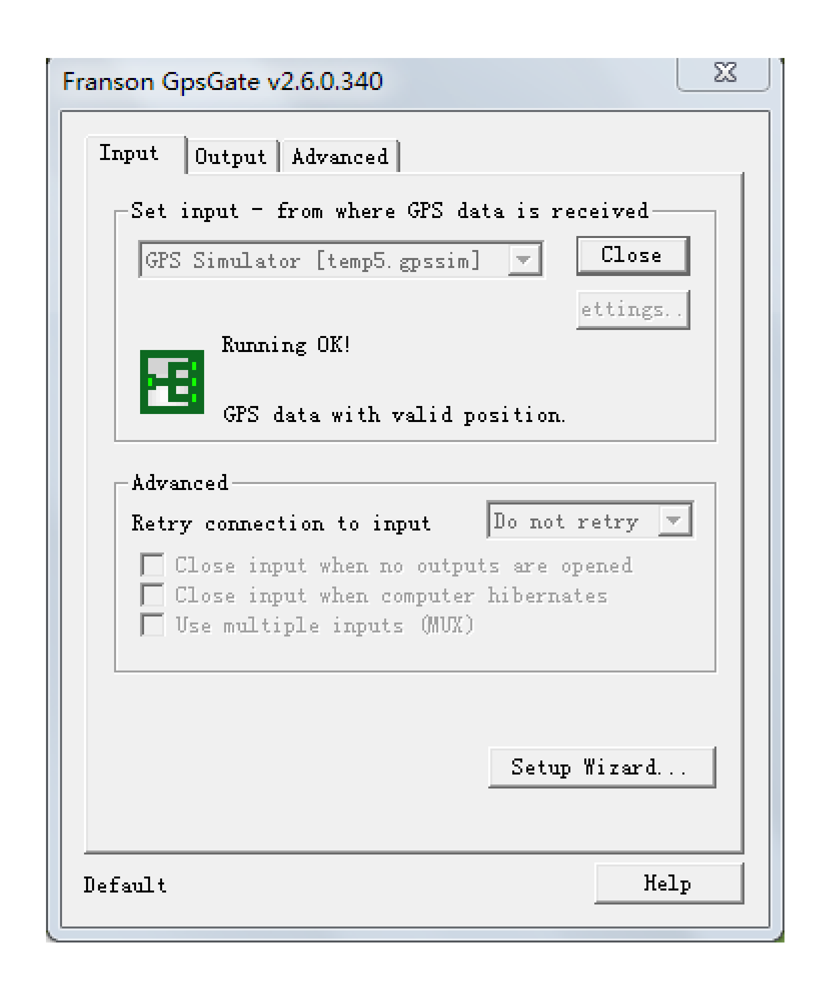


图 1 GPS GATE 运行截图

2. Input 的 Tab 选择 GPS Simulator 后，Setting 按钮设定路线的经纬度，见图 2。

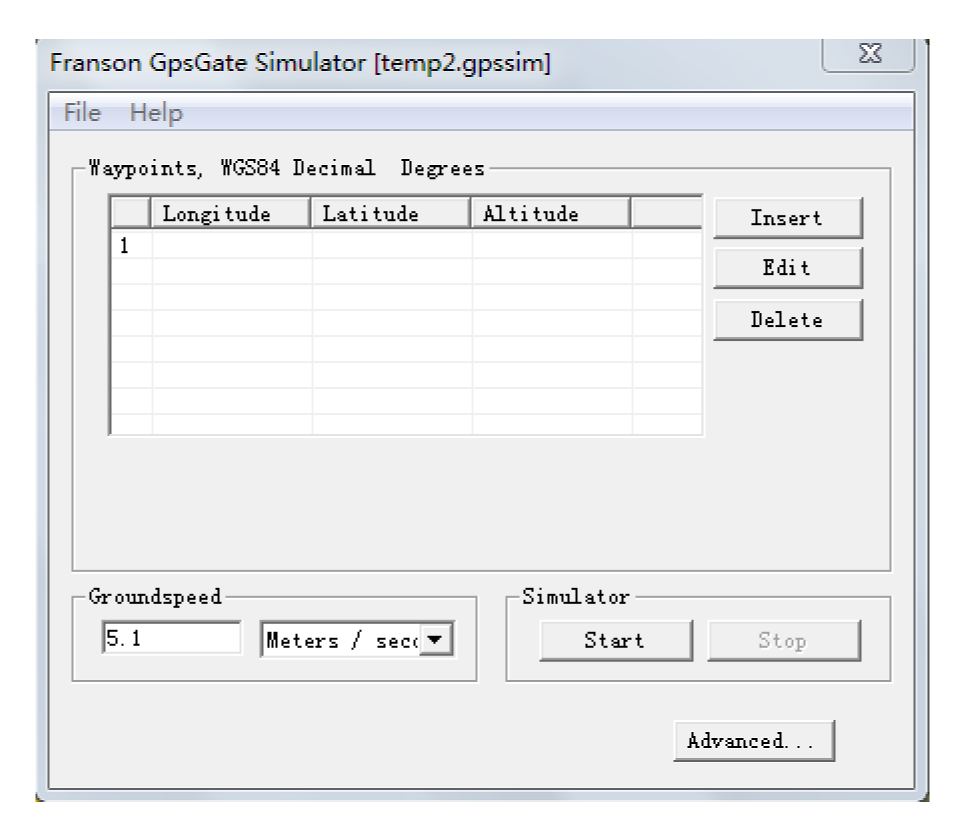


图 2 GPS GATE 设定路径界面

3. 再按下 OPEN 按钮才能以设定路线产生 NMEA 数据。
4. Output，选择输出到那个虚拟串口（Virtual COM），里入设定成虚拟串口 COMX 然后点击 Start，可以在 COMX 上接收到 GPS 数据。（见图 3）